
Barramento CoreConnect

MO801 – 1º semestre de 2006
Prof. Rodolfo Jardim de Azevedo

Fabiana Bellette Gil - RA 028671

CoreConnect

- Agenda
 - Conceitos básicos
 - Introdução ao CoreConnect
 - Arquitetura
 - Referências

Conceitos básicos

- **Barramento** - caminho de comunicação entre diversos dispositivos. Meio de transmissão compartilhado, onde são usados diversos fios elétricos para a ligação entre os dispositivos.
- **ASIC** - Application-Specific Integrated Circuit, um chip criado para uma aplicação particular
- **Core** – bloco que realiza alguma tarefa baseada em operações lógicas. Ex: processadores, DSPs, etc
- **IP-cores** (Intellectual Property) – cores sobre os quais as empresas mantêm direitos de propriedade intelectual, comercializando licença de uso
- Os padrões baseados em barramentos já são utilizados há algum tempo pelos mais diversos vendedores de IP-cores. Uma vantagem desta abordagem é que os cores desenvolvidos com interfaces para determinado barramento, podem ser adicionados diretamente à sistemas que usam o barramento apropriado.

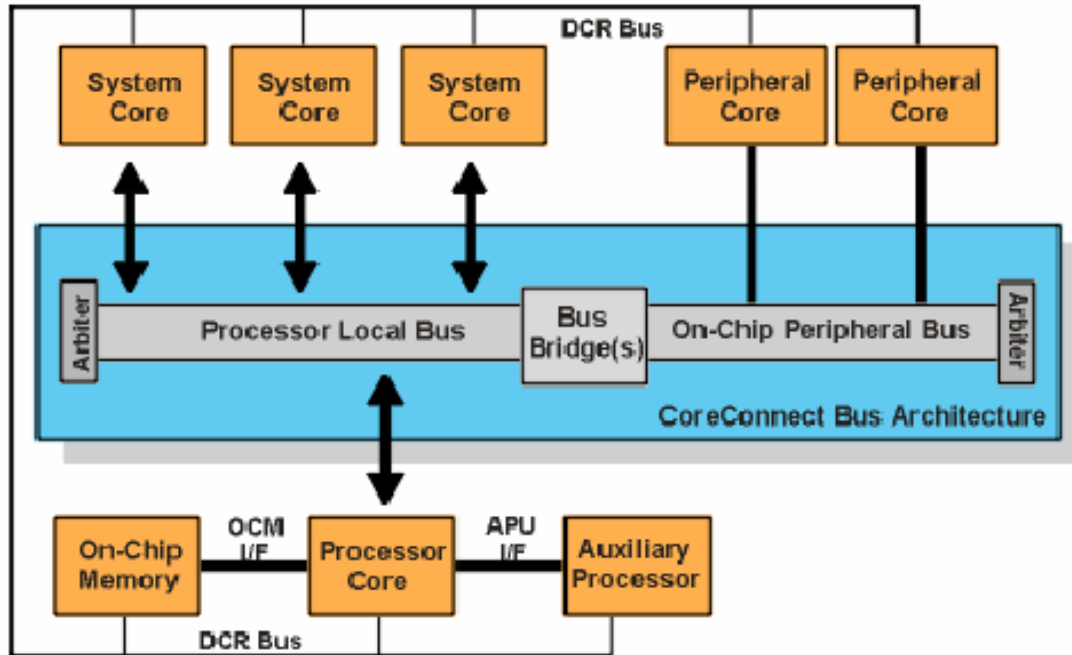
CoreConnect

- *CoreConnect* é uma iniciativa da IBM para prover integração e reutilização de processadores e *cores* de aplicações específicas em um só dispositivo (SoC).
- Inicialmente criado para suportar cores PowerPC para ASICs da IBM, passou a ser usado com outros processadores.
- A arquitetura do barramento *CoreConnect* permite aos projetistas de hardware montar projetos de SoCs facilmente, desde que o façam utilizando *cores* projetados com as especificações do barramento.

Arquitetura

- A arquitetura é composta por :
 - um barramento de processador local
(PLB - *Processor Local Bus*)
 - um barramento periférico interno ao circuito integrado
(OPB - *On-chip Peripheral Bus*)
 - um circuito que une esses dois barramentos
(*Bus Bridge*)
 - e um barramento de controle de dispositivo
(DCR - *Device Control Register*).

Arquitetura



Arquitetura

- Periféricos podem ser conectados ao PLB com grande largura de banda e baixa latência. Ex. processador, memória, controladores DMA.
- *Cores* com frequências mais baixas são conectados ao OPB, reduzindo o tráfego no PLB, o que aumenta o desempenho geral do sistema.

PLB

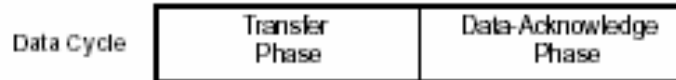
- O PLB soluciona questões de alto desempenho e baixa latência para *cores* integrados ao SoC.
- Ele é totalmente síncrono, e suporta a conexão de até 16 *cores* atuando como mestres e múltiplos *cores* escravos, cujo número varia de acordo com a implementação do árbitro.
- Existem 3 diferentes versões de arquitetura, com 32, 64 ou 128 bits.
- A especificação do PLB descreve uma arquitetura de sistema com detalhamento de sinais e transações.

Protocolo de Transferências PLB



- Uma transação PLB é formada de um ciclo de endereço e um ciclo de dados.
- Ciclos de endereços tem três fases: requisição, transferência e aceite.
 - Mestre envia endereço e sinais de transferência e requisita posse do barramento – request phase
 - Assim que consegue a posse, os sinais do mestre são apresentados ao escravo – transfer phase
 - Numa operação normal, o ciclo de endereço termina quando o escravo valida os sinais recebidos – acknowledge phase

Protocolo de Transferências PLB

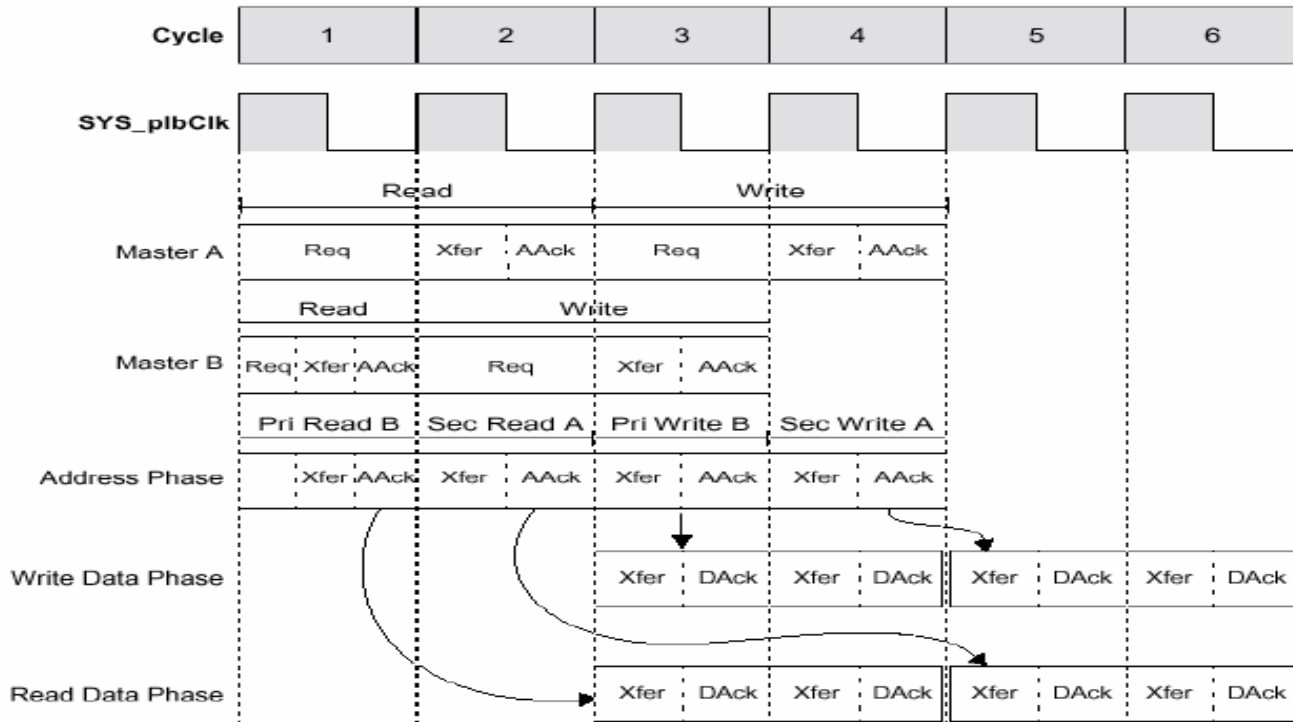


- Ciclos de dados tem duas fases: transferência e aceite.
 - Cada envio de dados tem as duas fases acima
 - Mestre irá usar barramento de escrita ou leitura para transferências
 - Sinais de aceite são necessários para cada envio de dados ocorrido
- Para um único envio de dados, o sinal de aceite indica o fim da transferência. Para transferências envolvendo vários blocos, o sinal de aceite é enviado para cada bloco, mas só indica fim do ciclo de dados depois de último bloco.

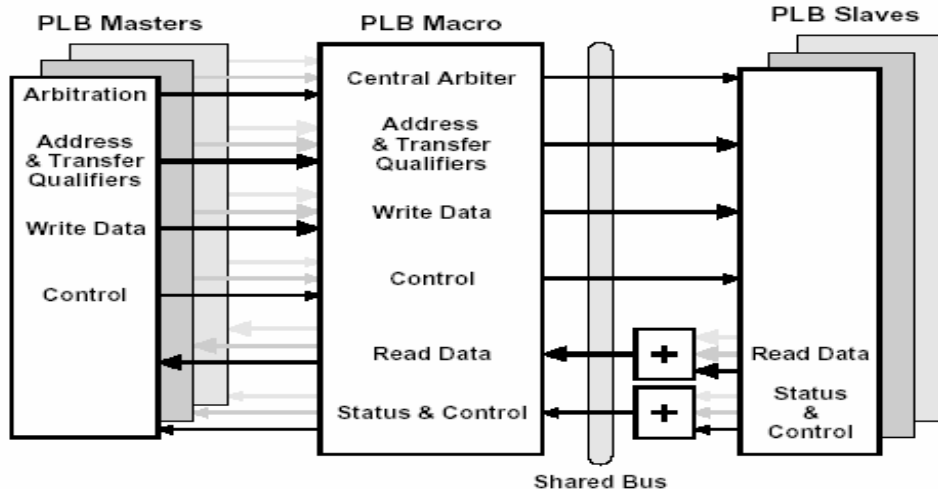
Transferência com sobreposição

- Barramentos de endereço, leitura e escrita são desacoplados, permitindo que os ciclos de endereço sejam sobrepostos com ciclos de leitura e escrita, assim como ciclos de leitura também sejam sobrepostos por ciclos de escrita. Dessa forma, é possível ter diferentes mestres utilizando os barramentos de dados e endereços.
- Pipeline permite que uma nova transferência seja iniciada antes da corrente terminar.

Transferência com sobreposição



PLB macro



Conexão de múltiplos mestres e escravos através do PLB macro.

Cada mestre PLB é ligado ao PLB macro através de barramentos de leitura, escrita e endereço separados e de uma infinidade de sinais de transferência. Os escravos PLB são ligados ao PLB macro através de barramentos compartilhados (mas desacoplados) de endereço, leitura e escrita e de vários sinais de controle de transferência e status para cada barramento de dados.

PLB macro

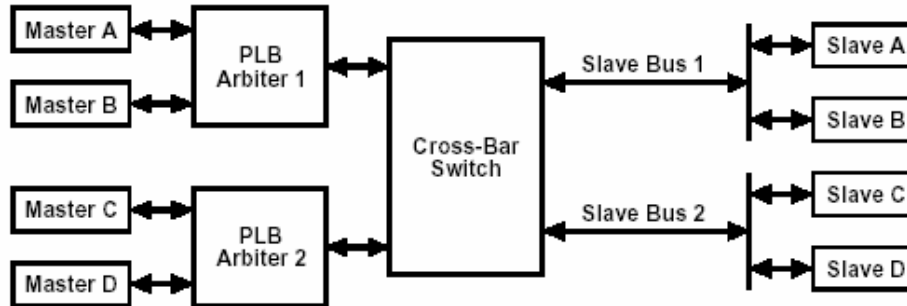
- O princípio de funcionamento baseia-se em um árbitro, que decide qual *core* de sistema (*System core*) vai fazer acesso ao barramento de cada vez. O mecanismo de arbitragem é flexível e permite implementação de vários esquemas de priorização. Suporta inclusive que o mestre utilize o barramento para transações atômicas através de locks.
- Cada barramento mestre conecta seus sinais de arbitragem, sinais de barramento de endereço e sinais de escrita e leitura para o árbitro que funciona como um grande multiplexador.

Desempenho do PLB

Performance features			
Feature	CoreConnect 32	CoreConnect 64	CoreConnect 128
PLB width	32-Bit	64-Bit	128-Bit
Max frequency	66 MHz	133 MHz	183 MHz *
Max bandwidth	264 MB/s	800 MB/s	2.9 GB/s *

* estimated

PLB Cross-Bar Switch



- Para grandes sistemas com múltiplos barramentos PLB CoreConnect, a IBM fornece o cross-bar-switch.
- Pode ser utilizado para permitir a comunicação entre mestres de um PLB e escravos de outro. No exemplo acima, foi colocado entre os árbitros PLB e os seus barramentos de escravos. Quando um master inicia uma transação, a CBS usa o endereço associado para selecionar o barramento escravo apropriado.
- Suporta transferências de dados simultâneos em ambos os barramentos PLB com um esquema de priorização para manipular requisições a um mesmo escravo. Adicionalmente, uma requisição de alta prioridade pode interromper uma de baixa prioridade.

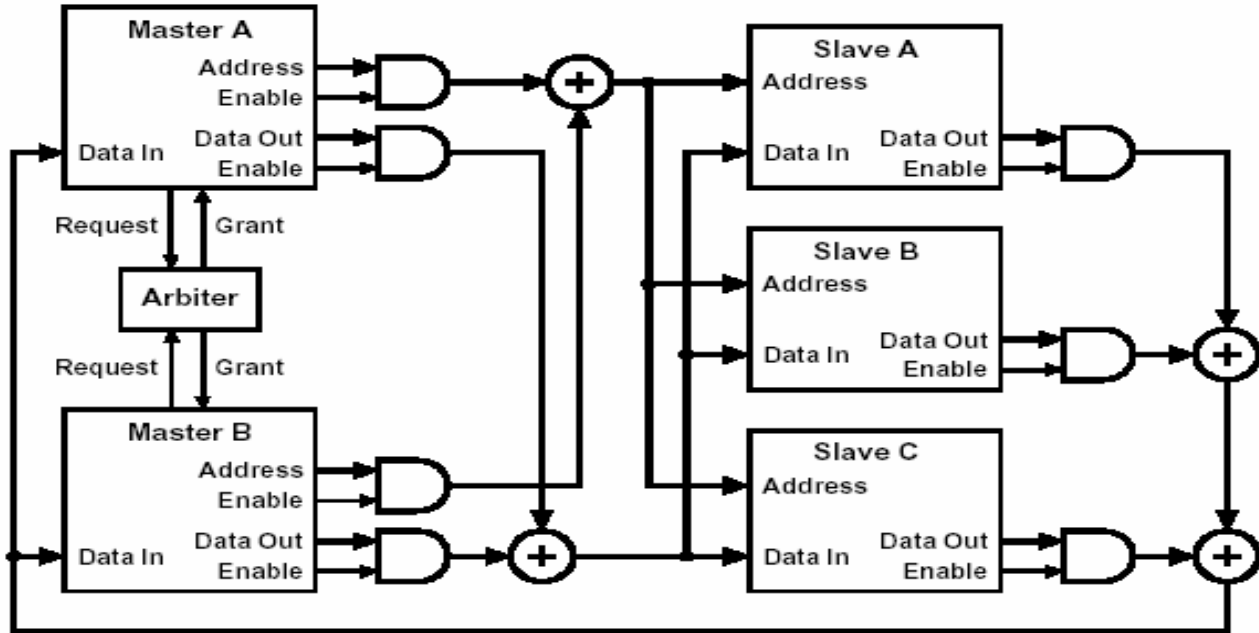
OPB

- O OPB é um barramento de entrada e saída, criado para aliviar gargalos de desempenho entre periféricos e o PLB. Exemplos de periféricos são portas paralelas, portas seriais, UARTs e outros dispositivos de banda estreita. Este barramento permite aos projetistas de sistema integrar facilmente periféricos ao ASIC. O OPB possui as seguintes características:
 - Protocolo totalmente síncrono, com barramentos para dados e endereços separados (cada qual com 32 bits);
 - Dimensionamento de barramento que permite transferências de bytes, palavras, e meias palavras;
 - Protocolo de endereçamento seqüencial (para modo rajada);
 - Inserção de ciclos de espera, para transferências de reduzida latência.

OPB

- Suporta múltiplos mestres e escravos pela implementação de barramentos de endereço e dados como um multiplexador distribuído.
- A figura mostra uma maneira de estruturar os barramentos de endereços e dados do OPB. Cada mestre é capaz de fornecer um endereço para os escravos, assim como todos os mestres e escravos são capazes de enviar e receber no barramento de dados.

Implementação física de um OPB



DCR – Device Control Register Bus

- Barramento controlado separadamente que liga todos os dispositivos, controladores e pontes.
- Caminho alternativo para monitorar e configurar registradores de controle individual
- A arquitetura do barramento DCR permite que transferências entre periféricos OPB ocorram independentemente, e concorrentemente, com transferências de dados entre o processador e memória ou entre outros dispositivos PLB.

OPB Bridge

- Mestres PLB acessam periféricos no barramento OPB através da OPB bridge. Ela age como um escravo para o PLB e como um mestre para o OPB. Ela suporta palavras (32 bits), meias-palavras (16 bits) e byte nas leituras e escritas em 32 bits no barramento OPB.
- Fornece dimensionamento dinâmico do tamanho do barramento, o que permite comunicação entre dispositivos de tamanhos diferentes.
- Quando o mestre (OPB bridge) solicita uma operação maior do que o escravo OPB, a ponte divide a operação em uma ou mais transferências menores.

CoreConnect x AMBA 2.0

	IBM CoreConnect Processor Local Bus	ARM AMBA 2.0 AMBA High-performance Bus
Bus Architecture	32-, 64-, and 128-bits Extendable to 256-bits	32-, 64-, and 128-bits
Data Buses	Separate Read and Write	Separate Read and Write
Key Capabilities	Multiple Bus Masters 4 Deep Read Pipelining 2 Deep Write Pipelining Split Transactions Burst Transfers Line Transfers	Multiple Bus Masters Pipelining Split Transactions Burst Transfers Line Transfers
	On-Chip Peripheral Bus	AMBA Advanced Peripheral Bus
Masters Supported	Supports Multiple Masters	Single Master: The APB Bridge
Bridge Function	Master on PLB or OPB	APB Master Only
Data Buses	Separate Read and Write	Separate or 3-state

Referências

- *CoreConnect bus architecture* Disponível em <http://www-03.ibm.com/chips/products/coreconnect/index.html> em 16/04/2006
- *The CoreConnect™ Bus Architecture* Disponível em [http://www-306.ibm.com/chips/techlib/techlib.nsf/techdocs/852569B20050FF77852569910050C0FB/\\$file/crcon_wp.pdf](http://www-306.ibm.com/chips/techlib/techlib.nsf/techdocs/852569B20050FF77852569910050C0FB/$file/crcon_wp.pdf) em 16/04/2006
- *CoreConnect™ Bus Architecture* Disponível em [http://www-306.ibm.com/chips/techlib/techlib.nsf/techdocs/852569B20050FF7785256991004DB5D9/\\$file/crcon_pb.pdf](http://www-306.ibm.com/chips/techlib/techlib.nsf/techdocs/852569B20050FF7785256991004DB5D9/$file/crcon_pb.pdf) em 16/04/2006
- Mesquita; Daniel Gomes, *Contribuições para reconfiguração parcial, remota e dinâmica de FPGAs* Disponível em http://www.inf.pucrs.br/~moraes/papers/dissertacao_mesquita.pdf em 16/04/2006